REMARKS/ARGUMENTS

These remarks are submitted in response to the Office Action of September 12, 2005 (Office Action). This response is filed after the 3-month shortened statutory period, and as such, a retroactive extension of time is hereby requested. The Examiner is authorized to charge the appropriate extension fee to Deposit Account 50-0951.

At page 1 of the Office Action, paragraph 2, Claims 1-13 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,502,131 to Vaid, *et al.* (hereinafter Vaid).

Independent Claims 1, 10 and 12 have each been amended to further emphasize certain aspects of Applicants' invention. Dependent claims 2-9, 11, and 13 have also been amended to emphasize certain aspects of the invention. The claims also have been amended to clarify that these claims recite subject matter within the technological art of creating telecommunication service components for use in advanced intelligent networks. These amendments are supported throughout the Specification, as described herein. No new matter is introduced by virtue of the amendments.

I. Applicants' Invention

Prior to addressing the rejections on the art, a brief review of the Applicants' invention is in order. Applicants' invention provides a visual tool which can be used to create service components for use in a service logic execution environment (SLEE) in an integrated network such as a JAIN-compliant network. The present invention mitigates the deficiencies of the prior art by providing a visual tool through which telecommunication service components can be created using drag-and-drop and smartguide techniques. Furthermore, by providing an intuitive graphical user interface through which service building block characteristics can be specified, a non-specialist using Applicants' invention can create service components for use in a SLEE without undertaking the tedious task of hand-coding such service components.

A visual tool that is provided for creating a JAIN-compliant telecommunication service component for use in a SLEE can include a first visual smartguide for creating JAIN-compliant service building blocks. Each service building block can include at least one JAIN-compliant telecommunication event handler for handling specific events received from an event routing bus in the SLEE. The visual tool also can include a second visual smartguide for creating deployment descriptors for the created JAIN-compliant service building blocks. Each deployment descriptor can include a JAIN-compliant service description and a list of supported events which can be handled by an associated service building block. Finally, the visual tool can include a visual composition interface through which visual representations of the JAIN-compliant service building blocks can be arranged to form the service component. Notably, the JAIN-compliant service building blocks can be software components, for example Java beans.

In one aspect of the visual tool, the first visual smartguide can include at least one selectable procedure for generating a plurality of telecommunication utility classes for inclusion in a JAIN-compliant service building block. In addition, the first visual smartguide can include a database of JAIN-compliant telecommunication event handlers from which the at least one event handler can be selected for addition to the service building block. Likewise, the second visual smartguide can include a database of JAIN-compliant telecommunication event handlers from which a list of supported events for inclusion in the deployment descriptor can be constructed. Finally, the visual tool can further include a service container which encapsulates the JAIN-compliant service component.

A method for visually generating a JAIN-compliant telecommunication service component can include specifying at least one JAIN-compliant service building block, wherein the specification includes visually selecting a plurality of JAIN-compliant telecommunication event handlers for inclusion in the at least one JAIN-compliant service building block. The method can also include exporting the at least one JAIN-

compliant service building block. The exporting step can produce a deployment descriptor which describes telecommunication events for which the at least one JAIN-compliant telecommunication service building block has been configured to handle. The at least one service block can be visually arranged, wherein the arrangement can form the service component. Finally, the JAIN-compliant telecommunication service component produced by the visual arrangement can be configured for insertion in a SLEE in an integrated network. Notably, in one aspect of the method, the method can further include the step of encapsulating the JAIN-compliant telecommunication service component in a service component container.

It should be appreciated by one of ordinary skill in the art and clear from the specification (pages 1-5 and FIG. 1) that JAIN is an open published standard, which includes a set of Java interfaces and tools that allow applications to provide value added services in telecommunication networks. A JAIN-compliant network includes a protocol layer, a signaling layer, an application layer, and a service layer. The SLEE is an environment within the service layer. JAIN-compliant service components within the SLEE can interact with underlying protocol stacks without having specific knowledge of the protocol stack. Accordingly, the JAIN-compliant service components can use the call model provided by the servicing layer to implement telephony services.

The claimed invention is a specific visual environment targeted at telecommunications applications based on the SLEE architecture. The visual SLEE environment is a complex, run-time, executable environment based on the SLEE specification that structures application logic of telecommunications applications as a collection of reusable object-orientated components. The SLEE specification further details requirements for composing these components into higher level and more sophisticated telecommunication services.

II. The Claims Define Over The Prior Art

Turning to the rejections on the art, claims 1-13 were rejected under 35 U.S.C. § 102(e) as being anticipated by Vaid. Briefly, Vaid discloses a method and system for monitoring or profiling a quality of service within one or more information sources in a network of computers. More particularly, the invention of Vaid relates to presenting a quality of service (QoS) within a network using traffic monitoring software which is distributed throughout the network. A QoS describes the throughput of a network and is often measured by responsiveness such as the amount of time spent waiting for data on a network. Notably, Vaid's invention provides a visual QoS management tool for performance monitoring and accounting of network traffic channel statistics. The visual management tool also allows for the control of incoming and outgoing traffic through the network at nodes containing the traffic monitoring software.

Effectively, Vaid teaches the creation of a visual application using distributed traffic monitoring programs to monitor the flow of packets within a network. Software traffic monitoring programs are deployed at nodes within the network to monitor traffic. The visual tool receives traffic reports from the nodes running the software programs and visually presents the traffic flow results in the form of charts within a graphical user interface. A user can change traffic policies using the visual tool to control the traffic behavior of traffic classes with the software programs deployed at the nodes within the network. For example, a user can adjust the flow of data within traffic classes such as FTP, HTTP, or PointCast services at a node within the network using the visual tool (Vaid, Col. 21, line 5).

It can be appreciated that Vaid's invention is a visual traffic tool for monitoring and managing a network QoS for existing data services. However, Vaid does not teach adding new functionality or accessibility for extending existing services. Vaid does not teach creating visual service building blocks or visually configuring extensions to new services. Understandably, Vaid accesses already existing services for informational reporting purposes and for adjusting the flow of information from already existing service elements. In contrast, Applicants' invention is directed towards exposing underlying

telecommunication protocols for providing new visual services through an open network Applications Programming Interface (API). In particular, Applicants' invention specifically implements an open network API through Java for Advanced Intelligent Network (JAIN) service programming for creating an extended JAIN-compliant telecommunication service component. The JAIN API implementation defines a call model that allows for the creation and extension of telecommunication service components outside of the PSTN switch environment. The service components can be shared or used by other network components desiring access to underlying media services such as Media Gateway Control Protocol (MGCP), Session Initiated Protocol (SIP), and Transactional Capabilities Application Part (TCAP) as well as protocols residing in higher layers of the telecommunications protocol stack.

Applicants' invention allows for JAIN applications or services operating at the service level to communicate directly with JAIN protocol adapters in a service logic execution environment (SLEE). Notably, Applicants' invention is concerned with implementing JAIN API's to open up underlying resource protocols thereby allowing SLEE applications access to previously unavailable data provided by the resource protocols. Applicants' invention makes telecommunication service components available to JAIN-compliant SLEE applications desiring access to the underlying resource protocols through a visual interface. In contrast to Vaid, Applicants do not deploy service components for receiving feedback for performance monitoring or control. Applicants' invention extends telecommunication service components for providing connectivity management and call control to the SLEE application implementing the service components.

Further, Vaid makes no mention of a SLEE or JAIN API implementation within the context of a real time execution environment. The teachings provided by Vaid do not apply within a real-time execution environment, for example, the passing of data and tasks from one JAIN service component to another in the SLEE. Vaid's teachings of a visual tool for traffic monitoring and management are not applicable to this type of

configurable and extendable execution environment. Vaid simply does not teach JAIN-compliant service components in a SLEE. Vaid's specification is tailored to a visual tool for network traffic monitoring and management, and which does not have real-time telephony timing and configuration concerns for JAIN-compliant telecommunication service components, and does not function within the same problem space as the Applicants' invention. For example, FIG. 13 of Vaid's invention corresponds to a visual interface for visualizing traffic bandwidth, whereas FIG. 7 of Applicants' invention corresponds to a drag and drop environment for extending JAIN telecommunication service components by rearranging the connection and configuration of JAIN service components.

Vaid fails to contemplate a SLEE or the extension of JAIN compliant telecommunication service components (real time components that execute in an operational environment) in any fashion. Vaid is silent with regards to a SLEE or with JAIN-compliant telecommunication service components that execute within a SLEE. The cited portion of Vaid (Col. 20 line 65-Col. 22 Line 15) details a visual tool for monitoring traffic for QoS through which a user can modify plots or charts of the traffic flow. In contrast, Applicants' invention provides a visual tool for accessing underlying protocol resources and modifying configurable parameters exposed by an implementation of the JAIN API for creating JAIN-compliant telecommunication service components. Moreover, Vaid provides no teachings regarding extensions to a JAIN enabled telecommunication service component comprising a visually arranged sequence of JAIN-compliant service building blocks for use in a SLEE that is explicitly claimed by the Applicants. Vaid does not teach a visual tool for creating or extending JAIN-compliant telecommunication service components for use in a service logic execution environment. Applicants also teach a visual smart guide comprising a database of JAIN-compliant telecommunication event handlers from which a list of supported JAIN telecommunication events can be constructed for inclusion in a deployment descriptor. Vaid does not teach a deployment descriptor for specifying events that a service building

block can handle or how the events can be serviced within a SLEE. In particular, Vaid does not teach the use of a class loader to identify configuration and loading parameters associated with a service component to be loaded. The class loader is necessary because it informs the service component as to the underlying resources that are available for which it would otherwise not be aware. In contrast, Vaid teaches deploying a traffic monitoring program such as a traffic class at network nodes to measure traffic flow for reporting back traffic information to the visual tool.

Vaid does not even contemplate a JAIN-compliant telecommunication service component that reconfigures itself based on information contained within a descriptor file of a traffic class. In fact, Vaid does not require a descriptor file to configure the traffic monitoring program, as the traffic monitoring program is a passive monitoring device. In contrast, Applicants' JAIN-compliant service component actively identifies underlying resources that are available for configuring itself within the SLEE. Understandably, Applicants' invention teaches that a service component can configure itself using a deployment descriptor when the component is unaware of the underlying JAIN protocol resources. Understandably, Applicants' virtual tool allows a user to identify and configure loading parameters that a JAIN-compliant service component can access to configure itself within a SLEE. The virtual tool provides configurable access to underlying resources and parameters exposed by the JAIN API for creating a JAIN-compliant service component.

Applicants' invention is used to arrange and configure JAIN-compliant service building blocks for receiving and transmitting telecommunication events to and from at least one JAIN configured protocol stack through a JAIN compliant signaling layer. Applicants have amended the claims to define elements of the SLEE architecture within the claims themselves. Applicants respectfully assert that the prior art fails to provide any suggestion, motivation, or teaching for Vaid's visual tool for creating a JAIN-compliant service component for use in a SLEE. Accordingly, Independent claims 1, 10, and 12 have been amended to emphasize these certain features of Applicants' invention.

Applicants respectfully assert, therefore, that the reference fails to teach each feature of amended independent Claims 1, 10 and 12, and that the claims thus define over the prior art. Applicants, moreover, respectfully assert that whereas the remaining claims each depend from one of the amended independent claims, the dependent claims likewise define over the prior art.

CONCLUSION

Applicants respectfully submit that, as described herein, Vaid fails to expressly or inherently teach each of the features recited in amended Claims 1 to 13. Applicants respectfully maintain, therefore, that each of Claims 1 through 13 define over the prior art. Applicants believe that this application is now in full condition for allowance, which action is respectfully requested. Applicants request that the Examiner call the undersigned if clarification is needed on any matter within this Amendment, or if the Examiner believes a telephone interview would expedite the prosecution of the subject application to completion.

Respectfully submitted,

Date: January 12, 2006

Gregory A. Nelson, Registration No. 30,577
Richard A. Hinson, Registration No. 47,652
Marc A. Boillot, Registration No. 56,164
AKERMAN SENTERFITT
Customer No. 40987
Post Office Box 3188
West Palm Beach, FL 33402-3188
Telephone: (561) 653-5000